NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

CONFIGURATION MANAGEMENT EVALUATION GUIDANCE FOR HIGH ROBUSTNESS SYSTEMS

by

Michael E. Gross

March 2004

Thesis Advisor:	Cynthia Irvine
Co-Advisor:	Tim Levin
Second Reader:	Nelson Irvine

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** March 2004	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE: Configuration Management Evaluation Guidance for High Robustness Systems			**5. FUNDING NUMBERS**
6. AUTHOR(S) Michael E. Gross			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

Configuration Management (CM) plays a vital role in the development of trusted computing systems. The Common Criteria (CC) provides a framework for performing Information Technology (IT) security evaluations of these systems and further emphasizes CM's role in the development and evaluation process by specifying a minimum set of CM qualities for any Evaluated Assurance Level (EAL). As an evaluation guide, the Common Methodology for Information Technology Security Evaluation, Part 2: Evaluation Methodology (CEM), recommends a minimum set of CM guidelines which can be used by evaluators in the performance of a CM evaluation at a given Evaluated Assurance Level (EAL). Evaluators and developers will quickly note the CEM's lack of recommended CM guidelines at the higher assurance levels.

Thorough study of the listed references supports the hypothesis for this work: Guidance extension of the CEM for high assurance CM is useful. As an assurance mechanism complete CM guidance helps users of high assurance products obtain a degree of confidence the system security requirements operate as intended and do not contain clandestine code. Complete CM guidance provides evaluators a "completed assurance scale" and ensures only authorized changes were made to the TOE during development.

14. SUBJECT TERMS Configuration Management, Common Criteria, Common Evaluation Methodology Guidelines, High Assurance, IT Product Evaluation			**15. NUMBER OF PAGES** 86
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

CONFIGURATION MANAGEMENT EVALUATION GUIDANCE FOR HIGH ROBUSTNESS SYSTEMS

Michael E. Gross
Lieutenant, United States Navy
B.A., University of Oklahoma, 1994

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
March 2004

Author: Michael E. Gross

Approved by: Cynthia Irvine
Thesis Advisor

Tim Levin
Co-Advisor

Nelson Irvine
Second Reader

Dan C. Boger
Chairman, Department of Information Sciences

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Configuration Management (CM) plays a vital role in the development of trusted computing systems. The Common Criteria (CC) provides a framework for performing Information Technology (IT) security evaluations of these systems and further emphasizes CM's role in the development and evaluation process by specifying a minimum set of CM qualities for each Evaluated Assurance Level (EAL). As an evaluation guide, the Common Methodology for Information Technology Security Evaluation, Part 2: Evaluation Methodology (CEM), recommends a set of minimum CM guidelines which can be used by evaluators in the performance of a CM evaluation at the lower Evaluated Assurance Levels. Evaluators and developers will quickly note the CEM's lack of recommended CM guidelines at the higher assurance levels.

Thorough study of the listed references supports the hypothesis for this work: Configuration Management guidelines are useful in the evaluation of trusted computing systems. As an assurance mechanism, complete CM guidance helps users of high assurance products obtain a degree of confidence the system security requirements operate as intended and do not contain clandestine code. Complete CM guidance provides evaluators with a "completed assurance scale" and ensures only authorized changes were made to the TOE during development.

Useful CM guidelines at the higher assurance levels (EAL5, 6, and 7) will help developers and evaluators ensure products meet the minimum requirements needed for high assurance systems.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to acknowledge Dr. Cynthia Irvine, Mr. Tim Levin (thesis advisors), and Dr. Nelson Irvine (second reader) for their outstanding support throughout the thesis process. My thesis "team" provided guidance when requested, support when needed, and direction as required and were instrumental throughout the thesis process.

In addition to my thesis "team", Mr. Ron Bottomly, National Security Agency, provided valuable evaluator feedback. His helpful insight helped me clarify my interpretation of the Common Criteria. I also wish to thank Mr. Daniel Faigin, of the Aerospace Corporation, for suggesting this topic.

I would also like to acknowledge my family. Their patient support cannot be emphasized enough. Melissa my wife, Stephanie and Sydney my daughters, and Michael my son graciously shared their time throughout my studies at NPS. I am grateful for all of their love and support they have given me.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Configuration Management (CM) is an engineering discipline that is widely used in various development and manufacturing environments. In the context of the Common Criteria, CM has dual roles. When viewed from the developer's standpoint, CM helps to prevent unauthorized modification to an information technology product during the development process. Evaluators "see" or "perceive" an alternate view of CM's role. In this context, a CM evaluation assists the consumer in identifying the evaluated product, and ensures all configuration items are uniquely identified and required procedures were used to track and control changes made to the product.

The Common Criteria provides an assurance scale based upon the premise that greater assurance results are achieved through more rigorous engineering paradigms accompanied by commensurately greater evaluation efforts. Within this framework, CM is one of seven assurance classes and is further divided into three assurance families. The assurance families provide the minimum requirements which must be met for a given evaluated assurance level (EAL).

The Common Evaluation Methodology (CEM) bridges the gap between abstract CC requirements and specific evaluator actions by providing minimum guidelines for evaluation at each Evaluated Assurance Level (EAL). However, the CEM provides no guidelines beyond EAL4 and does not address the evaluation of CM in high assurance systems. Thorough study of the listed references supports the hypothesis for this work: Configuration Management guidelines are useful in the evaluation of trusted computing systems. As an assurance mechanism, complete CM guidance helps users of high assurance products obtain a degree of confidence the system security requirements operate as intended and do not contain clandestine code. Complete CM guidance provides evaluators with a "completed assurance scale" and ensures only authorized changes were made to the TOE during development

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

Configuration Management is an engineering discipline that is widely used in various manufacturing environments for controlling changes made to products under development. As an assurance mechanism in the development of Information Technology (IT) systems, Configuration Management helps to provide confidence that a product or system will operate as intended, and does not accidentally or intentionally deviate from its approved design specifications. This work explores Configuration Management and its relationship to trusted computing systems.

B. OBJECTIVES

The Common Criteria for Information Technology Security Evaluation contains security specifications for IT products/systems [CCP199]. Through a mutual recognition arrangement among several countries, it was established that the Common Criteria (CC) would be utilized as the common basis for performing IT evaluations. As a companion document to the CC, the Common Methodology for Information Technology Security Evaluation [CEM99] provides guidelines for the application of CC requirements in the evaluation of products at lower Evaluated Assurance Levels (EALs): EAL1-EAL4.

This work develops guidelines for higher assurance levels, specifically: EAL5, EAL6, and EAL7.

C. GENERATING GUIDELINES

This thesis focuses on a primary question, "Are Configuration Management evaluation methodology guidelines useful?" While an obvious short answer is "yes", a need for guidelines should be substantiated before generating guidelines. If a need exists, then supporting documentation is required in order to generate useful guidelines. Requirements from the CC [CCP399] and previous CEM work [CEM99] provide the supporting documentation needed to generate CM evaluation methodology guidelines for higher assurance levels.

D. ORGANIZATION OF THESIS

Configuration Management Evaluation Guidance for High Robustness Systems contains six chapters. Chapter II gives the reader general background on Configuration

Management, setting the stage for understanding what type of guidelines are appropriate for the evaluation of high assurance CM requirements. Chapter III relates configuration management and high assurance within the Common Criteria framework. A need for evaluation guidelines and the associated requirements for meeting those guidelines are constructed in Chapter IV. Recommended configuration management evaluation guidelines constructed utilizing academic research on the Common Criteria, prior evaluation guidelines, the developer's CM system, and requirement dependencies are presented in Chapter V. Chapter VI highlights the need for further work which will help maintain guideline validity.

II. CONFIGURATION MANAGEMENT

Configuration management is widely used in several disciplines and job descriptions. By understanding configuration management's origin, definition, purpose, and value the reader should be able to understand its use and relation to the Common Criteria.

A. ORIGIN OF CONFIGURATION MANAGEMENT (CM)

Configuration Management (CM) describes an engineering discipline developed in the late 1950s and 1960s as a means to control the skyrocketing costs of complicated hardware and mechanical systems [ICM96]. As a discipline, CM continues to evolve and currently encompasses many different industries and is implemented through automated and non-automated tools and processes. Due to widespread use, CM can mean different things to different people based on an organization's culture and processes. From a personnel perspective the term might address the roles mangers play when performing CM duties or functions. A product view might relate CM as a description of the product's components and sub-assemblies that make up the components. Another example includes the use of CM as a means for process improvement. This view, espoused by the Capability Maturity Model (CMM), an organizational scheme with five hierarchical "levels", requires CM processes to be implemented and in use before an organization can reach CMM level 2. (People, Product, and Process perspectives were mentioned in the introduction section, page xli-xliii, [CMP03]). While there might be many other interpretations, for the purpose of this work, CM describes the control process related to software or computer system product development.

B. DEFINITION OF CM

While CM can have numerous associations, a definition of CM as it relates to the Common Criteria (CC) and Common Evaluation Methodology (CEM) has not been formalized by inclusion within the CC. Numerous references outside of the CC provide useful definitions of CM. Certainly, a good definition would include a derivation from standards documents used prior to acceptance of the CC. An obvious choice for a definition source would have to include a definition from previous evaluation literature [AGT88]:

> Configuration Management maintains control of a system throughout its life-cycle, ensuring that the system in operation is the correct system, implementing the correct security policy.

While this may be helpful it is important to realize the guideline from which this definition is taken has been superceded by the CC. Another definition from current work in software configuration management is provided by Ann Mette Jonassen Hass [CMP03]:

> Configuration management is the unique identification, controlled storage, change control and status reporting of selected intermediate work products, product components, and products during the life of a system.

Key elements for any CM definition used by evaluators should recognize identification and control as concepts used in the majority of CM definitions in references today.

C. PURPOSE OF CM

CM has two roles within the CC and CEM. From a developer standpoint the CC [CCP399] describes CM's purpose as:

> Configuration management (CM) helps to ensure that the integrity of the TOE (Target of Evaluation) is preserved, by requiring discipline and control in the processes of refinement and modification of the TOE and other related information. CM prevents unathorised modifications, additions, or deletions to the TOE, thus providing assurance that the TOE and documentation used for evaluation are the ones prepared for distribution.

From an evaluator standpoint the CEM [CEM99] describes CM's role as:

> The purpose of the configuration management activity is to assist the consumer in identifying the evaluated TOE, to ensure that configuration items are uniquely identified, and the adequacy of the procedures that are used by the developer to control and track changes that are made to the TOE. This includes details on what changes are tracked, how potential changes are incorporated, and the degree to which automation is used to reduce the scope for error.

D. VALUE OF CM

The value of CM lies in its inherent abilities in providing a control mechanism able,

> …to maintain the integrity of products as they evolve from specifications through design, development, and production [ICM96].

In terms of the Common Criteria (CC), maintaining this integrity helps to provide an assurance (grounds for confidence) that the products meets their security objectives [CCP399]. By ensuring the developer has met the required CM security assurance requirements, the evaluator is providing an independent review on behalf of consumers who might not have the expertise to perform an objective evaluation.

Understanding CM's origin helps to show how the discipline has been incorporated into various engineering activities and industries. Despite wide use, most definitions from past references and current software engineering material provide identification and control of configuration items as key elements for any relevant CM definition. In addition these concepts are echoed in CM's purpose as described in the CC and CEM. The value of CM lies in its ability to maintain the integrity of products throughout the development process. By understanding CM's origin, definition, purpose, and value we will be able to explore its role within high assurance systems.

THIS PAGE INTENTIONALLY LEFT BLANK

III. CM'S ROLE IN HIGH ASSURANCE SYSTEMS

Prior to understanding CM's role in high assurance systems, a short definition and explanation of assurance and its necessity are required. Additionally, background information on the CC assurance approach will also guide efforts in determining CM's role in the CC and the CEM. Finally, how CM requirements change as the assurance level changes will highlight the need for guidelines as a way to help developers and evaluators understand changing CM requirements as the evaluated assurance level changes. The need for this understanding provides support for this work's hypothesis: guidance extension of the CEM for high assurance CM is useful.

A. DEFINITION OF ASSURANCE

Before CM's role within high assurance systems is explored, it is necessary to understand what is meant by assurance and "high assurance" within the CC. The CC defines assurance as [CCP199]:

> …grounds for confidence that an entity meets its security objectives.

Further definition clarification comes from Part 3 of the CC [CCP399] and states:

> Assurance is grounds for confidence that an IT product or system meets its security objectives.

If assurance provides grounds for confidence, it would follow that high assurance provides grounds for high confidence that an IT product or system meets its security requirements. In broader context, the assurance provided during an evaluation helps to demonstrate specific security requirements in a Security Target (ST) or general requirements in a Protection Profile (PP) are met [CCP199].

B. NEED FOR HIGH ASSURANCE

The ultimate requirement for high assurance can be determined by an asset's value by its owner. Frequently high value assets are also of value to attackers who wish to compromise or damage the asset. Most security courses determine damage through loss of confidentiality, loss of integrity, or loss of availability [SMS03]. Any IT product or system that places high value on these areas has a need for high security requirements in order to minimize the damage (loss of value) to the asset. Users of high security

products need a high degree of confidence the system security requirements operate as intended and do not contain clandestine code [SCS80]. This confidence can be achieved by way of a high assurance evaluation of the system. With respect to evaluation assurance levels articulated in the CC, "high" assurance is often related to Evaluated Assurance Level (EAL) 5 and above. Alternatively, the term "high robustness" is used in various CC related documents currently in production. High robustness refers to EAL levels "greater than 4".

C. CC ASSURANCE APPROACH

The CC acknowledges many approaches can be used to provide assurance, however the CC focuses on "traditional" evaluation approaches as a means of gaining assurance. Some of the evaluation assurance techniques listed in the CC are [CCP399]:

a) analysis and checking of process(es) and procedure(s);

b) checking that process(es) and procedure(s) are being applied;

c) analysis of the correspondence between TOE design representations;

d) analysis of the TOE design representation against the requirements;

e) verification of proofs;

f) analysis of guidance documents;

g) analysis of functional tests developed and the results provided;

h) independent functional testing;

i) analysis for vulnerabilities (including flaw hypothesis);

j) penetration testing.

D. SECURITY ASSURANCE SCALE

The CC provides an assurance scale based on the premise that greater assurance results will be achieved through more rigorous engineering paradigms accompanied by commensurately greater evaluation efforts. This graded assurance scale specifies the minimum effort required to achieve the desired assurance level. Advancing to the next higher assurance level is based on three elements [CCP399]:

a) scope — that is, the effort is greater because a larger portion of the IT product or system is included;

b) depth — that is, the effort is greater because it is deployed to a finer level of design and implementation detail;

c) rigour — that is, the effort is greater because it is applied in a more structured, formal manner.

While developers can state their product meets a certain level, the individual product can only be at a rated Evaluated Assurance Level (EAL) after a certifying authority has verified an independent evaluation of the product. Only after an evaluation is certified, can a product state it has an EAL value. The CC's assurance scale ranges from EAL1 to EAL7, with EAL7 being the highest possible assurance level.

E. CM'S DEVELOPMENT ROLE IN THE CC

CM is one of seven classes of security assurance requirements designated within the CC framework. As indicated by its name, the CM class contains requirements for CM assurance in the development context. These requirements enable the developer to provide evidence of Target of Evaluation (TOE) integrity and how it was maintained during the development process. The CM class is composed of three assurance families: CM automation, CM capabilities, and CM scope. Each assurance family can be further broken down into at least one assurance component. Every assurance component contains a set of assurance elements. The CC defines an assurance element [CCP399] as the:

…smallest security requirement recognized in the CC.

As the smallest recognized security requirement, assurance elements belong to one of three groupings: developer, content and evaluator. CM's development role in high assurance systems comes from required developer assurance elements as required by Part 3 of the CC. Additionally, developers are also given content requirements which are required to provide TOE integrity evidence.

F. CM'S EVALUATION ROLE IN THE CC

Just as CM maintains a development role in the CC framework, it also provides an evaluation role. As mentioned above, Part 3 of the CC provides security assurance requirements. The smallest recognized evaluator assurance element describes [CCP399]:

> …the activities that shall be performed by the evaluator. This set of actions explicitly includes confirmation that the requirements prescribed in the content and presentation of evidence elements have been met.

The CC also provides explicit actions and analysis that shall be performed in addition to those already performed by the developer. When accomplished by an evaluator, the required evaluator element is used to make a determination that the provided content and developer elements satisfy the minimum assurance requirements for a stated EAL.

G. CM REQUIREMENTS CHANGE AS THE EVALUATED ASSURANCE LEVEL (EAL) INCREASES

By now, it should be easily recognized that requirements will change as the EAL level changes. Figure 1 provides a portion of Table B.1 [CCP399] from Part 3 of the CC and provides a quick reference which can be used by developers and evaluators to determine what component version numbers by assurance family are needed at a given EAL. Since our focus is only on CM, only that part of the table has been provided.

Table B.1 - Evaluation assurance level summary

Assurance Class	Assurance Family	Assurance Components by Evaluation Assurance Level						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
Configuration management	ACM_AUT				**1**	1	**2**	2
	ACM_CAP	**1**	**2**	**3**	**4**	4	**5**	5
	ACM_SCP			**1**	**2**	**3**	3	3
Delivery and	ADO_DEL		**1**	1	2	2	2	**3**

Figure 1. Portion of Table B.1 [CCP399]

While Figure 1 provides a ready reference for version numbers, it is not a substitute for an exact understanding of "what changes" as the EAL increases. Thorough study of Part 3 of the CC [CCP399] is needed to ensure developers and evaluators understand exact changes when moving from one EAL to another. It is important to remember the EAL assurance scale is graded and represents the minimum requirements to be met in order to provide the stated assurance level. For each EAL the requirements either inherit previous assurance family components or build on previous components by

including additions and/or changes to meet the minimum requirements for the next higher level. Visually, these concepts are presented in Figure 2, and exact CM requirement changes are presented in the following section.

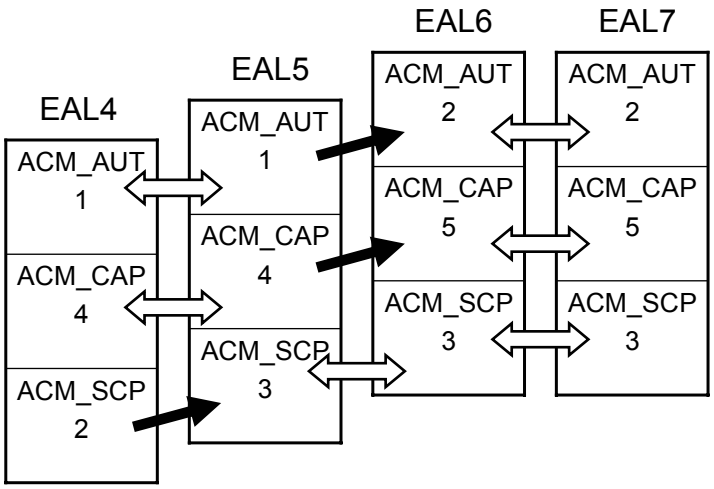

Figure 2. CM Component Change and Inheritance as the EAL Increases

H. CM CHANGES AS THE EAL INCREASES

For the un-initiated developer or evaluator, Figure 2 provides a visual representation for change and inheritance regarding CM assurance requirements as the EAL increases. As mentioned in the previous section, an increase in EAL could require changes or additions to previous requirements within an assurance family. These changes or additions are represented with a solid slanted arrow (e.g. changes to the scope family (ACM_SCP) are needed when increasing from EAL4 to EAL5). Figure 2 represents inheritance from one EAL to another by using double arrows (e.g. EAL5 automation requirements are inherited without change or additions from EAL4). Only the exact requirements changes or additions are presented in the subsections below.

1. CM Changes from EAL4 to EAL5

Scope assurance family (ACM_SCP) objectives change to include development tools CM coverage [CCP399]:

> Development tools play an important role in ensuring the production of a quality version of the TOE. Therefore, it is important to control modifications to these tools.

The following change to content and presentation of evidence elements is noted:

ACM_SCP.2.1C maps to ACM_SCP.3.1C and adds development tools and related information to the minimum list of items that must be tracked by the CM system [CCP399]:

> ACM_SCP.3.1C The CM documentation shall show that the CM system, as a minimum, tracks the following: the TOE implementation representation, design documentation, test documentation, user documentation, administrator documentation, CM documentation, security flaws, and development tools and related information.

2. CM Changes from EAL5 to EAL6

Automation (ACM_AUT) family requirements change in order to meet the minimum assurance for EAL6. Automation objectives change to include the following objective [CCP399]:

> Providing an automated means of ascertaining changes between versions of the TOE and identifying which configuration items are affected by modifications to other configuration items assists in determining the impact of the changes between successive versions of the TOE. This in turn can provide valuable information in determining whether changes to the TOE result in all configuration items being consistent with one another.

Content and presentation of evidence elements requirements change to include a change and two additional requirements. ACM_AUT.1.1.C maps to ACM_AUT.2.1.C and includes a change that adds all other configuration items [CCP399]:

> ACM_AUT.2.1C The CM system shall provide an automated means by which only authorized changes are made to the TOE implementation representation, and to all other configuration items.
>
> ACM_AUT.2.5C The CM system shall provide an automated means to ascertain the changes between the TOE and its previous version.
>
> ACM_AUT.2.6C The CM system shall provide an automated means to identify all other configuration items that are affected by the modification of a given configuration item.

In addition to automation changes, Capability (ACM_CAP) assurance family changes are also needed to meet EAL6 requirements. The capabilities objectives change to include advanced support [CCP399]:

> Integration procedures help to ensure that generation of the TOE from a managed set of configuration items is correctly performed in an authorized manner.
>
> Requiring that the CM system be able to identify the master copy of the material used to generate the TOE helps to ensure that the integrity of this material is preserved by the appropriate technical, physical and procedural safeguards.

Content and presentation of evidence elements requirements change to include a change and nine additional requirements. ACM_CAP.5.3C inherits all of the requirements from ACM_CAP.4.3C and adds integration procedures as required CM documentation for EAL 5 [CCP399]:

> ACM_CAP.5.3C The CM documentation shall include a configuration list, a CM plan, an acceptance plan, and integration procedures.
>
> ACM_CAP.5.13C The integration procedures shall describe how the CM system is applied in the TOE manufacturing process.
>
> ACM_CAP.5.14C The CM system shall require that the person responsible for accepting a configuration item into CM is not the person who developed it.
>
> ACM_CAP.5.15C The CM system shall clearly identify the configuration items that comprise the TSF.
>
> ACM_CAP.5.16C The CM system shall support the audit of all modifications to the TOE, including as a minimum the originator, date, and time in the audit trail.
>
> ACM_CAP.5.17C The CM system shall be able to identify the master copy of all material used to generate the TOE.
>
> ACM_CAP.5.18C The CM documentation shall demonstrate that the use of the CM system, together with the development security measures, allow only authorized changes to be made to the TOE.
>
> ACM_CAP.5.19C The CM documentation shall demonstrate that the use of the integration procedures ensures that the generation of the TOE is correctly performed in an authorized manner.

ACM_CAP.5.20C The CM documentation shall demonstrate that the CM system is sufficient to ensure that the person responsible for accepting a configuration item into CM is not the person who developed it.

ACM_CAP.5.21C The CM documentation shall justify that the acceptance procedures provide for an adequate and appropriate review of changes to all configuration items.

3. CM Changes from EAL6 to EAL7

As noted in Part 3 of the CC, there are no changes or additions to CM in order to meet EAL7 requirements for CM.

An understanding of assurance and why it is important are key concepts when considering CM's role in high assurance systems. The assurance approach utilized within the CC focuses on a graded scale in which the minimum items needed for an associated assurance level are expressed as the minimum requirements for an EAL. As an assurance class, CM and its families play an important role as an assurance mechanism. By presenting the CM changes needed to meet the next higher level, the CC provides developers and evaluators a roadmap that can be used when developing or evaluating an IT product or system.

IV. CONFIGURATION MANAGEMENT EVALUATION METHODOLOGY GUIDELINES

Guidelines can be defined as statements or other indications of policy or procedure by which to determine a course of action. While the previous sections give background and CM requirements, they do not to provide useful guidelines that would enable developers and evaluators to incorporate needed requirements in order to meet a stated EAL. As a complementary document to the CC, the *Common Methodology for Information Technology Security Evaluation Part 2: Evaluation Methodology* (CEM) bridges the gap between requirements and performing an evaluation of a system intended to meet those requirements.

A. ARE CONFIGURATION MANAGEMENT (CM) GUIDELINES USEFUL IN THE EVALUATION OF TRUSTED SYSTEMS?

As a course of action, the CEM provides a set of guidelines for use by the international IT security evaluation community. As a companion document to the CC, the CEM guidelines help evaluators determine what actions are needed to determine that a TOE has met the requirements for an EAL. It is important to note, however, that the CEM only offers guidelines through EAL4 and leaves higher assurance guidelines to be developed by individual countries. CM Guidelines for EAL5 – EAL7 are beneficial for several reasons:

- EAL 5 – 7 CM guidelines would provide a "ready reference" training guide for evaluators;
- CM guidelines emphasize what minimum actions need to be performed in the context of a particular EAL;
- CM guidelines provide a plain language CC companion document;
- CM guidelines will help prepare U.S. organizations perform an EAL7 evaluation.

B. NEED FOR GUIDELINES

A need for a complete set of EAL guidelines is highlighted by recent congressional testimony. In his September 17, 2003, testimony before the Committee on Government Reform Subcommittee on Technology, Information Policy, Intergovernmental Relations and the Census, Edward Roback, Chief, Computer Security Division National Institute of Standards and Technology mentions the application of the

CC provides needed assurance for the government's IT products [ECC03]. His testimony also states that ways are needed to reduce the costs of evaluations and provides examples of "what could be done". Specific examples from his testimony follow:

> Develop CC interpretations that clarify and simplify how parts of the CC are to be evaluated.
>
> Provide better training to lab evaluators and NIAP (National Information Assurance Partnership) validators, with emphasis on which actions need to be performed and which do not.
>
> Provide an extensive/complete set of guidance documents for all stakeholders in the evaluation process (e.g., developers, evaluators, validators, commercial and government users).
>
> Provide clear guidance to stakeholders to choose only those assurance requirements that are meaningful for their intended use/environments.
>
> Perform a critical assessment of the current evaluation process to ensure that:
>
> - NIAP activities and levels of effort are consistent with those of other CC Recognition Arrangement partners
> - Evaluation activities are being performed efficiently
> - There are no unnecessary activities being performed
> - All activities that can be performed in parallel are in fact done that way.

While Mr. Roback's testimony represents a broad based "call to arms", a complete set of evaluation guidelines answers his call for more research and provides: better training to evaluators, plain language interpretations of the CC, and a more complete set of guidance to the CC documents. Clearly a complete set of CEM guidelines which include EAL5 through 7 are appropriate in answering his call.

Currently, the CEM's usefulness is limited only to the lower EAL levels. CEM guidelines which include higher assurance levels could help to standardize evaluations performed at higher levels within the United States. As the need for higher level evaluations increases, it is appropriate for the U.S. CM community to develop, promote,

and recommend minimum guidelines for IT products which need evaluation at higher assurance levels which are not currently included in the CEM.

C. GUIDELINE FORMAT

EAL5 through 7 guidelines developed here pick up where the CEM stops. Due to inheritance and change properties of CM requirements as the EAL increases, prior guideline work from EAL4 [CEM99] will be used as a starting point for generating new guidelines. As a follow on to the CEM, the proposed guidelines are intended to help define the minimum effort for achieving an evaluation at a given EAL and:

> …to provide guidance on ways and means to accomplish the evaluation [CEM99].

Although guidelines can take many forms, the proposed guidelines are meant to present a recommended action item/CM check list which evaluators can incorporate into any test plan. Similar to the CEM, the guidelines will provide the content and presentation of evidence elements individually, followed by recommended evaluator action items for the required element. Where practical, the content and presentation elements will be numbered and language similar to the CEM will be used.

D. EAL5 GUIDELINES

EAL5 guidelines are only useful when the context of the assurance level is understood [CCP399]:

> This EAL represents a meaningful increase in assurance from EAL4 by requiring semiformal design descriptions, the entire implementation, a more structured (and hence analyzable) architecture, covert channel analysis, and improved mechanisms and/or procedures that provide confidence that the TOE will not be tampered with during development.

In support of the increased assurance, the CM assurance class scope family components change to include the maximum scope attainable. These requirements directly support confidence that the TOE will not be tampered with during development. New developmental tamper proofing efforts include a change to the list of items tracked by the CM system. ACM_SCP.3.1C covers this change [CCP399]:

> The CM documentation shall show that the CM system, as a minimum, tracks the following: the TOE,…,security flaws, and development tools and related information.

1. EAL5 Action Item Generation

Careful reading of the new requirement points to three exploration areas when determining the need for recommending minimum action items for an evaluator performing an evaluation of ACM_SCP.3.1.C: the CM documentation at EAL5, the CM system, and the application notes.

a. Reviewing the CM Documentation

CM documentation at EAL5 is comprised of items noted in ACM_CAP.4.3C [CCP399]:

> ACM_CAP.4.3C The CM documentation shall include a configuration list, a CM plan and an acceptance plan.

What these items are and do follow:

> ACM_CAP.4.4C The configuration list shall describe the configuration items that comprise the TOE.
>
> ACM_CAP.4.7.C The CM plan shall describe how the CM system is used.
>
> ACM_CAP.4.12.C The acceptance plan shall describe the procedures used to accept modified or newly created configuration items as part of the TOE.

Previous changes in scope (EAL3 to 4) have set the precedent that the evaluator "shall check" the configuration list to ensure it includes the minimum set of items tracked by the CM system. However, due to the importance of development tools and the need for prevention of subversion during the development process, it would be appropriate for evaluators to confirm if the provided development tools and related documentation on the configuration list are sufficient for the required evaluated assurance level. Clearly, development tools may be listed in other areas of CM documentation. As a point of clarity, to "confirm" [CCP399] means to:

> …indicate that something needs to be reviewed in detail, and that an independent determination of sufficiency needs to be made.

Reviewing CM documentation would lead to two recommendations for evaluator action items in support of ACM_SCP.3.1.C: build on the inherited ACM_SCP.2.1C and also require the evaluator to "confirm" necessary visibility within all of the CM documentation.

b. CM System Operation

"Vaporware" is a term often used to describe non-existent software. While we would like to trust every developer, it is important to verify the configuration list items presented in the CM documentation directly correspond to the configuration items within the CM system. Verification between documentation and software will ensure the developer is not providing a "vapor list". As noted previously, "confirm" means to make an independent determination of sufficiency. This would not require the evaluator to have independent access to the CM system. As such this action item could be easily verified by observation of the appropriate portion of the CM system.

c. Application Notes

The application notes for ACM_SCP.3.1C help to provide the evaluator examples of development tools and related documentation [CCP399]:

> ACM_SCP.3.1C introduces the requirement that development tools and other related information be tracked by the CM system. Examples of development tools are programming languages and compilers. Information pertaining to TOE generation items (such as compiler options, installation/generation options, and build options) is an example of information relating to development tools.

2. Recommended Evaluator Actions

By examining the application notes, the CM documentation and the CM system, the following three action items are recommended. Together, the inherited items from EAL4 and the recommended additions constitute the proposed evaluator CM guidelines for EAL5 (bold type shows the difference between EAL4 and EAL5 guidelines):

ACM_SCP.3.1C

- The evaluator shall check that the configuration list includes the minimum set of items required by the CC to be tracked by the CM system. The list should include at a minimum:
  - all documentation required to meet the target level of assurance;
  - test software (if applicable);

- the TOE implementation representation (i.e. the components or subsystems that compose the TOE). For a software-only TOE, the implementation representation may consist solely of source code; for a TOE that includes a hardware platform, the implementation representation may refer to a combination of software, firmware and a description of the hardware (or a reference platform).
- the documentation used to record details of reported security flaws associated with the implementation (e.g. problem status reports derived from a developer's problem reporting database).
- **development tools (e.g. programming languages and compilers) and related documentation (e.g. information pertaining to TOE generation items (such as compiler options, installation/generation options, and build options).**

- **The evaluator shall confirm the level of documentation necessary for ensuring proper visibility of the development tools and related documentation throughout the entire CM documentation.**
- **The evaluator shall confirm the development tools and related documentation listed in the CM documentation are resident in the CM system. In performance of this action item, the evaluator does not need independent access to determine the sufficiency of correspondence between the system configuration list and documentation.**

3. EAL5 Summary

Reasonable CM guideline recommendations have been derived from the application notes, CC requirements, and the vendor's CM system. These recommended evaluator actions help to ensure the TOE will not be tampered with during the development process.

E. EAL6 GUIDELINES

Similar to EAL5, the context for EAL6 must be understood before guidelines can be recommended [CCP399]:

> This EAL represents a meaningful increase in assurance from EAL5 by requiring more comprehensive analysis, a structured representation of the implementation, more architectural structure (e.g. layering), more comprehensive independent vulnerability analysis, systematic covert channel identification, and improved configuration management and development environment controls.

CM system "automation" and "capabilities" changes are key in providing the minimum assurance requirements for an evaluation at EAL6. Although the ideas of inheritance and change have been mentioned in other thesis sections, EAL 6 recommended guidelines for automation and capabilities are evolved from the CEM [CEM99]. EAL6 requires no further changes to CM "scope" family and those items are inherited from EAL5.

1. EAL6 Action Item Generation in Support of Complete CM Automation

EAL6 action item generation for complete CM automation (ACM_AUT) follows a similar investigative path used to generate evaluator action items for EAL5. As presented in Section III, the CM automation changes at EAL6 include [CCP399]:

> ACM_AUT.2.1C The CM system shall provide an automated means by which only authorized changes are made to the TOE implementation representation, and to all other configuration items.
>
> ACM_AUT.2.5C The CM system shall provide an automated means to ascertain the changes between the TOE and its preceding version.
>
> ACM_AUT.2.6C The CM system shall provide an automated means to identify all other configuration items that are affected by the modification of a given configuration item.

By exploring the CM documentation, CM system, application notes and dependencies on other assurance families, evaluator action items for the automation requirements can be determined.

a. Reviewing the CM Documentation

At first glance, it might appear there is no need for reviewing CM documentation in support of recommending automation evaluator guidelines at EAL6. However complete automation is dependent on the authorization controls required by the CM capabilities family (ACM_CAP.3 version numbers), and should have been established at EAL3. Authorization controls requires a configuration list and a CM plan [CCP399]. The case for reviewing the CM plan for documenting complete automation can be implied from the dependency on the content and presentation elements for (ACM_CAP.3), specifically:

ACM_CAP.3.7C The CM plan shall describe how the CM system is used.

ACM_CAP.3.8C The evidence shall demonstrate that the CM system is operating in accordance with the CM Plan.

The CM plan provides the written description of how authorized changes will be made to the TOE and configuration items. The requirement for automation is intended to reduce the potential for error in complex development environments. Any "automated means" regarding system operation will be described by the CM plan in order to support the dependent items above.

b. CM System Operation

Complete CM system automation will be described by the CM plan. Operation of the CM system will help the evaluator determine if the "automated means" mentioned in the CM plan is adequate and accurately represents the requirements or procedures in the CM plan. Various CM systems may implement the "automated means" differently. An example of determining changes between the TOE and a preceding version may include some sort of report (paper or screen shot). The evaluator should "confirm" the CM system provides the "automated means" to ascertain the TOE changes or configuration item changes by observing the appropriate operational functions of the CM system.

c. Application Notes

A single application note for complete CM automation would help to provide an example which can be mentioned when recommending an evaluator action item to support ACM_AUT.2.5C [CCP399]:

ACM_AUT.2.5C introduces a requirement that the CM system provide an automated means to ascertain the changes between the TOE and its preceding version. If no previous version of the TOE exists, the developer still needs to provide an automated means to ascertain the changes between the TOE and a future version of the TOE.

d. Exploring Family Dependencies

Family dependencies are an important part for understanding automation requirements at EAL6. Evaluation of complete automation depends on the authorization controls put in place when meeting ACM_CAP.3 version numbers. As mentioned in the documentation section above, the CM plan will describe the authorization procedures that

will be implemented by automated means at EAL6. An evaluator should have a complete understanding of ACM_CAP.3 prior to commencing an evaluation at EAL6.

2. EAL6 Action Item Generation for Advanced Support

CM capabilities (ACM_CAP) changes which require evaluation action items are [CCP399]:

> ACM_CAP.5.3C The CM documentation shall include a configuration list, a CM plan, an acceptance plan, and Integration procedures.
>
> ACM_CAP.5.13C The integration procedures shall describe how the CM system is applied in the TOE manufacturing process.
>
> ACM_CAP.5.14C The CM system shall require that the person responsible for accepting a configuration item into CM is not the person who developed it.
>
> ACM_CAP.5.15C The CM system shall clearly identify the configuration items that comprise the TSF.
>
> ACM_CAP.5.16C The CM system shall support the audit of all modifications to the TOE, including as a minimum the originator, date, and time in the audit trail.
>
> ACM_CAP.5.17C The CM system shall be able to identify the master copy of all material used to generate the TOE.
>
> ACM_CAP.5.18C The CM documentation shall demonstrate that the use of the CM system, together with the development security measures, allow only authorized changes to be made to the TOE.
>
> ACM_CAP.5.19C The CM documentation shall demonstrate that the use of the integration procedures ensures that the generation of the TOE is correctly performed in an authorized manner.
>
> ACM_CAP.5.20C The CM documentation shall demonstrate that the CM system is sufficient to ensure that the person responsible for accepting a configuration item into CM is not the person who developed it.
>
> ACM_CAP.5.21C The CM documentation shall justify that the acceptance procedures provide for an adequate and appropriate review of changes to all configuration items.

a. Reviewing the CM Documentation

Integration procedures round out the fully developed CM documentation area which can be explored when recommending minimum evaluator guidelines at EAL6.

ACM_CAP.5.13C provides the following description of integration procedures [CCP399]:

> The integration procedures shall describe how the CM system is applied in the TOE manufacturing process.

Fully developed CM documentation is needed to support the requirements at EAL6. At this level any "system shall" verbiage in a requirement ultimately requires the system to perform an operation that meets a required operation or performs a function using information from the CM documentation. At this level a "one-for-one" documentation check is appropriate for recommending minimum evaluator action items to meet the capability requirements at EAL6.

b. CM System Operation

CM system capabilities provide advanced support at EAL6. As noted from the new capability requirements mentioned in Section 2 above (ACM_CAP.5.14C/5.15C/5.16C/5.17C), a minimum of four action items are recommended in order for evaluators to "confirm" the vendor's CM system performs the capabilities as intended.

c. Application Notes

There are no application notes to explore for CM capabilities.

d. Exploring Assurance Class Dependencies

Dependencies between the CM assurance (ACM) and Life cycle support assurance (ALC) classes occur at EAL6. While some CM specialists may feel the ALC requirements are different and beyond the scope of CM evaluation, it is appropriate to derive evaluator action items implied from the dependency between the two classes.

Content and presentation of evidence elements for life cycle support follow [CCP399]:

> ALC_DVS.2.1C The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.
>
> ALC_DVS.2.2C The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.

ALC_DVS.2.3C The evidence shall justify that the security measures provide the necessary level of protection to maintain the confidentiality and integrity of the TOE.

In addition to content and presentation of evidence, an additional evaluator action item from ALC_DVS should be considered [CCP399]:

The evaluator shall confirm that the security measures are being applied.

3. Summary for Generating EAL6 Guidelines

Successful evaluations at EAL 6 will require much greater effort (than EAL5) due to the need for fully developed automation and capabilities requirements imposed on the vendor's CM system. Family dependencies at EAL6 further aggravate difficult evaluation tasks faced by CM evaluators. By exploring the CM documentation, CM system, application notes, and family dependencies, the proposed evaluator guidelines supporting changes or additions at EAL6 also incorporate the ideas of change and inheritance from previous CEM and EAL5 work. EAL6 guidelines supporting new requirements follow (bold type shows the difference between EAL5 and EAL6):

ACM_AUT.2.1C

- The evaluator shall check the CM plan for a description of the automated measures to control access to the TOE implementation representation.
- The evaluator shall check the automated access control measures to determine that they are effective in preventing unauthorized modification of the TOE implementation representation.
- The evaluator shall check the CM plan for a description of the automated measures to control access to the configuration items.
- **The evaluator shall examine the automated access control measures to determine that they are effective in preventing unathorised modification of configuration items.**
    - The evaluator reviews the configuration management documentation to identify those individuals or roles authorized to make changes to the TOE implementation representation. For example, once it is under configuration management, access to an element of the implementation representation may only be allowed for the individual who performs the software integration role. Similarly, the evaluator reviews the CM documentation to identify those individuals or roles authorized to access a configuration item.

- The evaluator should exercise the automated access control measures to determine whether they can be bypassed by an unauthorized role or user. This determination need only be comprised of a few basic tests. Tests should cover the TOE implementation representation and configuration items.

ACM_AUT.2.5C

- **The evaluator shall check the CM documentation for a description of the automated means by which changes can be ascertained between the TOE and its preceding version.**
- **The evaluator shall examine the CM system to ensure it has the capability to provide automated tracking for changes between the TOE and its preceding version.**
  - **Previous TOE versions may or may not exist. Regardless, evaluators should exercise the automated change tracking within the CM system or verify the capability exists.**

ACM_AUT.2.6C

- **The evaluator shall check that the CM documentation includes information on the automated means to identify all other configuration items that are affected by the modification of a given configuration item.**
- **The evaluator shall examine the automated means used by the CM system to identify all other configuration items that are affected by the modification of a given configuration item.**
  - **The CM documentation should state how the automated means will identify the changes to configuration items that are affected by the modification of a given configuration item. CM systems may differ in the way how this information is presented**.

ACM_CAP.5.3C

- The evaluator shall check that the CM documentation provided includes a configuration list.
  - A configuration list identifies the items being maintained under configuration control.
- The evaluator shall check that the CM documentation provided includes a CM plan.
- The evaluator shall check that the CM documentation provided includes an acceptance plan.
- **The evaluator shall check that the CM documentation provided includes integration procedures.**

ACM_CAP.5.13C

- **The evaluator shall check that the integration procedures describe how the CM system is applied in the TOE manufacturing process.**

ACM_CAP.5.14C

- **The evaluator shall check that the CM documentation describes configuration item acceptance procedures.**
- **The evaluator shall verify configuration item acceptance procedures in the CM system perform as described by the CM documentation.**
    - **The CM system should prevent the person who developed the configuration item from being able to accept it.**

ACM_CAP.5.15C

- **The evaluator shall verify the CM documentation clearly identifies all configuration items that comprise the TSF.**
- **The evaluator shall verify that the configuration items that comprise the TSF are clearly identified in the CM system.**

ACM_CAP.5.16C

- **The evaluator shall verify the CM documentation describes the audit capabilities of the CM system.**
    - **The CM system, at a minimum, will track all TOE modifications to include: the originator, date, and time in the audit trail.**
- **The evaluator shall verify the CM system's TOE audit capabilities.**
    - **The audit portion of the CM system shall provide, at a minimum, the originator, date and time in the audit trail.**

ACM_CAP.5.17C

- **The evaluator shall verify the CM documentation describes how the CM system will identify the master copy of all material used to generate the TOE.**
- **The evaluator shall verify the CM system identifies the master copy of all material used to generate the TOE.**

ACM_CAP.5.18C

- **The evaluator shall verify the CM documentation describes how the use of the CM system and developmental security measures will only allow authorized changes to be made to the TOE.**

ACM_CAP.5.19C

- **The evaluator shall verify that the CM documentation describes how the use of the integration procedures will ensure that the generation of the TOE is performed correctly in the authorized manner.**

ACM_CAP.5.20C

- **The evaluator shall verify that the CM documentation sufficiently describes how the CM system will ensure that the person responsible for accepting a configuration item into the CM system is not the person who developed it.**

ACM_CAP.5.21C

- **The evaluator shall verify that the CM documentation provides adequate justification that the acceptance procedures provide adequate and appropriate review for all changes made to configuration items.**

ALC_DVS Dependency

- **The evaluator shall confirm that CM-specific security measures from the TSF are implemented and documented as necessary in the CM documentation.**

F. EAL 7 GUIDELINES

As noted in the Common Criteria, no additional CM requirements or changes are needed to perform an evaluation at EAL7. While no mandated requirements exist, some additional effort should be considered.

1. Increasing the Sample Size

Sampling is [CEM99]:

> … a defined procedure of an evaluator whereby some subset of a required set of evaluation evidence is examined and assumed to be representative for the entire set.

While sampling is allowed for some CEM work items through EAL4, the "blanket" recommendation for the recommended higher EALs has not been made. Although possibly obvious, it is appropriate for evaluators to determine the need for increased sample sizes in order to successfully perform evaluations at higher assurance levels. In most instances, high assurance products should have larger sample sizes than the recommended minimum size of 20%. Sampling sizes for high assurance products need not follow the [CEM99]:

…commensurate with cost effectiveness…

guideline. Evaluators should ensure the required level of effort is expended to ensure maximum scope, depth, and rigor are used when evaluating products at EAL7. Without firm sampling size requirements, evaluators must balance the competing needs of cost-effective evaluations and providing TOE consumers a level of confidence (from the act of performing a "high assurance" evaluation). At the highest assurance level (EAL7), it is appropriate to recommend 100% sampling sizes where possible. Simply put, smaller sample sizes mean more of the TOE has not been inspected. Increasing the TOE sampling percentage provides greater assurance that accidental or intentional TOE modifications have not occurred during the development process.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONFIGURATION MANAGEMENT EVALUATION GUIDELINES (EAL4 THROUGH EAL7)

Configuration Evaluation Methodology Guidelines provide a starting point for the evaluation process and where needed evaluators should seek further clarification within the CC and CEM as appropriate.

A. CM GUIDELINES FOR EAL4

It should be noted that the following guidelines were taken directly from the CEM [CEM99]. They are presented here in order to provide a starting point where the CEM leaves off.

ACM_AUT.1.1C

- The evaluator ***shall check*** the CM plan for a description of the automated measures to control access to the TOE implementation representation.
- The evaluator ***shall examine*** the automated access control measures to determine that they are effective in preventing unauthorised modification of the TOE implementation representation.
    - The evaluator reviews the configuration management documentation to identify those individuals or roles authorised to make changes to the TOE implementation representation. For example, once it is under configuration management, access to an element of the implementation representation may only be allowed for the individual who performs the software integration role.
    - The evaluator should exercise the automated access control measures to determine whether they can be bypassed by an unauthorised role or user. This determination need only comprise a few basic tests.

ACM_AUT.1.2C

- The evaluator ***shall check*** the CM documentation for automated means to support generation of the TOE from its implementation representation.
    - In this work unit the term *generation* applies to those processes adopted by the developer to progress the TOE from its implementation to a state ready to be delivered to the end customer.
    - The evaluator should verify the existence of automated generation support procedures within the CM documentation.
- The evaluator ***shall examine*** the automated generation procedures to determine that they can be used to support generation of the TOE.

- The evaluator determines that by following the generation procedures a TOE would be generated that reflects its implementation representation. The customer can then be confident that the version of the TOE delivered for installation implements the TSP as described in the ST. For example, in a software TOE this may include checking that the automated generation procedures help to ensure that all source files and related libraries that are relied upon to enforce the TSP are included in the compiled object code.
- It should be noted that this requirement is only to provide support. For example, an approach that placed Unix makefiles under configuration management should be sufficient to meet the aim, given that in such a case automation would have made a significant contribution to accurate generation of the TOE. Automated procedures can assist in identifying the correct configuration items to be used in generating the TOE.

ACM_AUT.1.3C

- The evaluator ***shall check*** that the CM plan includes information on the automated tools used in the CM system.

ACM_AUT.1.4C

- The evaluator ***shall examine*** the information relating to the automated tools provided in the CM plan to determine that it describes how they are used.
  - The information provided in the CM plan provides the necessary detail for a user of the CM system to be able to operate the automated tools correctly in order to maintain the integrity of the TOE. For example, the information provided may include a description of:
    - the functionality provided by the tools;
    - how this functionality is used by the developer to control changes to the implementation representation;
    - how this functionality is used by the developer to support generation of the TOE.
- The evaluator ***shall examine*** the CM system to determine that the automated tools and procedures described in the CM plan are used.
  - This work unit may be viewed as an additional activity to be carried out in parallel with the evaluator's examination into the use of the CM system required by ACM_CAP. The evaluator looks for evidence that the tools and procedures are in use. This should include a visit to the development site to witness operation of the

tools and procedures, and an examination of evidence produced through their use.

- For guidance on site visits see Annex B.5.

ACM_CAP.4.1C

- The evaluator ***shall check*** that the version of the TOE provided for evaluation is uniquely referenced.
  - The evaluator should use the developer's CM system to validate the uniqueness of the reference by checking the configuration list to ensure that the configuration items are uniquely identified. Evidence that the version provided for evaluation is uniquely referenced may be incomplete if only one version is examined during the evaluation, and the evaluator should look for a referencing system that is capable of supporting unique references (e.g. use of numbers, letters or dates). However, the absence of any reference will normally lead to a fail verdict against this requirement unless the evaluator is confident that the TOE can be uniquely identified.
  - The evaluator should seek to examine more than one version of the TOE (e.g. during rework following discovery of a vulnerability), to check that the two versions are referenced differently.

ACM_CAP.4.2C

- The evaluator ***shall check*** that the TOE provided for evaluation is labelled with its reference.
  - The evaluator should ensure that the TOE contains a unique reference such that it is possible to distinguish different versions of the TOE. This could be achieved through labelled packaging or media, or by a label displayed by the operational TOE. This is to ensure that it would be possible for consumers to identify the TOE (e.g. at the point of purchase or use).
  - The TOE may provide a method by which it can be easily identified. For example, a software TOE may display its name and version number during the start up routine, or in response to a command line entry. A hardware or firmware TOE may be identified by a part number physically stamped on the TOE.
- The evaluator ***shall check*** that the TOE references used are consistent.
  - If the TOE is labelled more than once then the labels have to be consistent. For example, it should be possible to relate any labelled guidance documentation supplied as part of the TOE to the evaluated operational TOE. This ensures that consumers can be confident that they have purchased the evaluated version of the TOE, that they have installed this version, and that they have the

correct version of the guidance to operate the TOE in accordance with its ST. The evaluator can use the configuration list that is part of the provided CM documentation to verify the consistent use of identifiers.

- The evaluator also verifies that the TOE reference is consistent with the ST.
- For guidance on consistency analysis see Annex B.3.

ACM_CAP.4.3C

- The evaluator ***shall check*** that the CM documentation provided includes a configuration list.
  - A configuration list identifies the items being maintained under configuration control.
- The evaluator ***shall check*** that the CM documentation provided includes a CM plan.
- The evaluator ***shall check*** that the CM documentation provided includes an acceptance plan.

ACM_CAP.4.4C

- The evaluator ***shall examine*** the configuration list to determine that it identifies the configuration items that comprise the TOE.
  - The minimum scope of configuration items to be covered in the configuration list is given by ACM_SCP.

ACM_CAP.4.5C

- The evaluator ***shall examine*** the method of identifying configuration items to determine that it describes how configuration items are uniquely identified.

ACM_CAP.4.6C

- The evaluator ***shall check*** that the configuration list uniquely identifies each configuration item.
  - The configuration list contains a list of the configuration items that comprise the TOE, together with sufficient information to uniquely identify which version of each item has been used (typically a version number). Use of this list will enable the evaluator to check that the correct configuration items, and the correct version of each item, have been used during the evaluation.

ACM_CAP.4.7C

- The evaluator ***shall examine*** the CM plan to determine that it describes how the CM system is used to maintain the integrity of the TOE configuration items.
    - The descriptions contained in a CM plan may include:
        - all activities performed in the TOE development environment that are subject to configuration management procedures (e.g. creation, modification or deletion of a configuration item);
        - the roles and responsibilities of individuals required to perform operations on individual configuration items (different roles may be identified for different types of configuration item (e.g. design documentation or source code));
        - the procedures that are used to ensure that only authorised individuals can make changes to configuration items;
        - the procedures that are used to ensure that concurrency problems do not occur as a result of simultaneous changes to configuration items;
        - the evidence that is generated as a result of application of the procedures. For example, for a change to a configuration item, the CM system might record a description of the change, accountability for the change, identification of all configuration items affected, status (e.g. pending or completed), and date and time of the change. This might be recorded in an audit trail of changes made or change control records;
        - the approach to version control and unique referencing of TOE versions (e.g. covering the release of patches in operating systems, and the subsequent detection of their application).

ACM_CAP.4.8C

- The evaluator ***shall check*** the CM documentation to ascertain that it includes the CM system records identified by the CM plan.
    - The output produced by the CM system should provide the evidence that the evaluator needs to be confident that the CM plan is being applied, and also that all configuration items are being maintained by the CM system as required by ACM_CAP.4.9C. Example output could include change control forms, or configuration item access approval forms.

- The evaluator ***shall examine*** the evidence to determine that the CM system is being used as it is described in the CM plan.
    - The evaluator should select and examine a sample of evidence covering each type of CM-relevant operation that has been performed on a configuration item (e.g. creation, modification, deletion, reversion to an earlier version) to confirm that all operations of the CM system have been carried out in line with documented procedures. The evaluator confirms that the evidence includes all the information identified for that operation in the CM plan. Examination of the evidence may require access to a CM tool that is used. The evaluator may choose to sample the evidence.
    - For guidance on sampling see Annex B.2.
    - Further confidence in the correct operation of the CM system and the effective maintenance of configuration items may be established by means of interview with selected development staff. In conducting such interviews, the evaluator should aim to gain a deeper understanding of how the CM system is used in practice as well as to confirm that the CM procedures are being applied as described in the CM documentation. Note that such interviews should complement rather than replace the examination of documentary evidence, and may not be necessary if the documentary evidence alone satisfies the requirement. However, given the wide scope of the CM plan it is possible that some aspects (e.g. roles and responsibilities) may not be clear from the CM plan and records alone. This is one case where clarification may be necessary through interviews.
    - It is expected that the evaluator will visit the development site in support of this activity.
    - For guidance on site visits see Annex B.5.

ACM_CAP.4.9C

- The evaluator ***shall check*** that the configuration items identified in the configuration list are being maintained by the CM system.
    - The CM system employed by the developer should maintain the integrity of the TOE. The evaluator should check that for each type of configuration item (e.g. high-level design or source code modules) contained in the configuration list there are examples of the evidence generated by the procedures described in the CM plan. In this case, the approach to sampling will depend upon the level of granularity used in the CM system to control CM items. Where, for example, 10,000 source code modules are identified in the configuration list, a different sampling strategy should be applied compared to the case in which there are only 5, or even 1.

The emphasis of this activity should be on ensuring that the CM system is being operated correctly, rather than on the detection of any minor error.

- For guidance on sampling see Annex B.2.

ACM_CAP.4.10C

- The evaluator ***shall examine*** the CM access control measures described in the CM plan to determine that they are effective in preventing unauthorised access to the configuration items.
  - The evaluator may use a number of methods to determine that the CM access control measures are effective. For example, the evaluator may exercise the access control measures to ensure that the procedures could not be bypassed. The evaluator may use the outputs generated by the CM system procedures and already examined as part of the work unit 4:ACM_CAP.4-13. The evaluator may also witness a demonstration of the CM system to ensure that the access control measures employed are operating effectively.
  - The developer will have provided automated access control measures as part of the CM system and as such their suitability may be verified under the component ACM_AUT.1

ACM_CAP.4.11C

- The evaluator ***shall check*** the CM documentation for procedures for supporting the generation of the TOE.
  - In this work unit the term *generation* applies to those processes adopted by the developer to progress the TOE from implementation to a state acceptable for delivery to the end customer.
  - The evaluator verifies the existence of generation support procedures within the CM documentation. The generation support procedures provided by the developer may be automated, and as such their existence may be verified under the component ACM_AUT.1.2C.
- The evaluator ***shall examine*** the TOE generation procedures to determine that they are effective in helping to ensure that the correct configuration items are used to generate the TOE.
  - The evaluator determines that by following the generation support procedures the version of the TOE expected by the customer (i.e. as described in the TOE ST and consisting of the correct configuration items) would be generated and delivered for installation at the customer site. For example, in a software TOE this may include checking that the procedures ensure that all

source files and related libraries are included in the compiled object code.

- The evaluator should bear in mind that the CM system need not necessarily possess the capability to generate the TOE, but should provide support for the process that will help reduce the probability of human error.

ACM_CAP.4.12C

- The evaluator ***shall examine*** the acceptance procedures to determine that they describe the acceptance criteria to be applied to newly created or modified configuration items.
  - An acceptance plan describes the procedures that are to be used to ensure that the constituent parts of the TOE are of adequate quality prior to incorporation into the TOE. The acceptance plan should identify the acceptance procedures to be applied:
    - at each stage of the construction of the TOE (e.g. module, integration, system);
    - to the acceptance of software, firmware and hardware components;
    - to the acceptance of previously evaluated components.
  - The description of the acceptance criteria may include identification of:
    - such configuration items;
    - any acceptance criteria to be applied before the configuration items are accepted (e.g. successful document review, or successful testing in the case of software, firmware or hardware).

ACM_SCP.2.1C

- The evaluator ***shall check*** that the configuration list includes the minimum set of items required by the CC to be tracked by the CM system.
  - The list should include the following as a minimum:
    - all documentation required to meet the target level of assurance;
    - test software (if applicable);
    - the TOE implementation representation (i.e. the components or subsystems that compose the TOE). For a software-only TOE, the implementation representation may consist solely of source code; for a TOE that includes a hardware platform, the implementation representation may

refer to a combination of software, firmware and a description of the hardware (or a reference platform).

- the documentation used to record details of reported security flaws associated with the implementation (e.g. problem status reports derived from a developer's problem reporting database).

ACM_SCP.2.2C

- The evaluator ***shall examine*** the CM documentation to determine that the procedures describe how the status of each configuration item can be tracked throughout the lifecycle of the TOE.
  - The procedures may be detailed in the CM plan or throughout the CM documentation. The information included should describe:
    - how each configuration item is uniquely identified, such that it is possible to track versions of the same configuration item;
    - how configuration items are assigned unique identifiers and how they are entered into the CM system;
    - the method to be used to identify superseded versions of a configuration item;
    - the method to be used for identifying and tracking configuration items through each stage of the TOE development and maintenance lifecycle (i.e. requirements specification, design, source code development, through to object code generation and on to executable code, module testing, implementation and operation);
    - the method used for assigning the current status of the configuration item at a given point in time and for tracking each configuration item through the various levels of representation at the development phase (i.e. source code development, through to object code generation and on to executable code, module testing and documentation);
    - the method used for identifying and tracking flaws relative to configuration items throughout the development lifecycle;
    - the method used for identifying correspondence between configuration items such that if one configuration item is changed it can be determined which other configuration items will also need to be changed.

- The analysis of the CM documentation for some of this information may have been satisfied by work units detailed under ACM_CAP.

B. CM GUIDELINES FOR EAL5

The following guidelines incorporate the changes and additions required in order to perform an evaluation at EAL5. Where appropriate, changes made to existing guidance from EAL4 [CEM99] are noted in bold face type.

ACM_AUT.1.1C

- The evaluator ***shall check*** the CM plan for a description of the automated measures to control access to the TOE implementation representation.
- The evaluator ***shall examine*** the automated access control measures to determine that they are effective in preventing unathorised modification of the TOE implementation representation.
    - The evaluator reviews the configuration management documentation to identify those individuals or roles authorized to make changes to the TOE implementation representation. For example, once it is under configuration management, access to an element of the implementation representation may only be allowed for the individual who performs the software integration role.
    - The evaluator should exercise the automated access control measures to determine whether they can be bypassed by an unathorised role or user. This determination need only comprise a few basic tests.

ACM_AUT.1.2C

- The evaluator ***shall check*** the CM documentation for automated means to support generation of the TOE from its implementation representation.
    - In this work unit the term *generation* applies to those processes adopted by the developer to progress the TOE from its implementation to a state ready to be delivered to the end customer.
    - The evaluator should verify the existence of automated generation support procedures within the CM documentation.
- The evaluator ***shall examine*** the automated generation procedures to determine that they can be used to support generation of the TOE.
    - The evaluator determines that by following the generation procedures a TOE would be generated that reflects its implementation representation. The customer can then be confident that the version of the TOE delivered for installation implements the TSP as described in the ST. For example, in a software TOE

this may include checking that the automated generation procedures help to ensure that all source files and related libraries that are relied upon to enforce the TSP are included in the compiled object code.

- It should be noted that this requirement is only to provide support. For example, an approach that placed Unix makefiles under configuration management should be sufficient to meet the aim, given that in such a case automation would have made a significant contribution to accurate generation of the TOE. Automated procedures can assist in identifying the correct configuration items to be used in generating the TOE.

ACM_AUT.1.3C

- The evaluator ***shall check*** that the CM plan includes information on the automated tools used in the CM system.

ACM_AUT.1.4C

- The evaluator ***shall examine*** the information relating to the automated tools provided in the CM plan to determine that it describes how they are used.
  - The information provided in the CM plan provides the necessary detail for a user of the CM system to be able to operate the automated tools correctly in order to maintain the integrity of the TOE. For example, the information provided may include a description of:
    - the functionality provided by the tools;
    - how this functionality is used by the developer to control changes to the implementation representation;
    - how this functionality is used by the developer to support generation of the TOE.
- The evaluator ***shall examine*** the CM system to determine that the automated tools and procedures described in the CM plan are used.
  - This work unit may be viewed as an additional activity to be carried out in parallel with the evaluator's examination into the use of the CM system required by ACM_CAP. The evaluator looks for evidence that the tools and procedures are in use. This should include a visit to the development site to witness operation of the tools and procedures, and an examination of evidence produced through their use.
  - For guidance on site visits see Annex B.5.

ACM_CAP.4.1C

- The evaluator ***shall check*** that the version of the TOE provided for evaluation is uniquely referenced.
    - The evaluator should use the developer's CM system to validate the uniqueness of the reference by checking the configuration list to ensure that the configuration items are uniquely identified. Evidence that the version provided for evaluation is uniquely referenced may be incomplete if only one version is examined during the evaluation, and the evaluator should look for a referencing system that is capable of supporting unique references (e.g. use of numbers, letters or dates). However, the absence of any reference will normally lead to a fail verdict against this requirement unless the evaluator is confident that the TOE can be uniquely identified.
    - The evaluator should seek to examine more than one version of the TOE (e.g. during rework following discovery of a vulnerability), to check that the two versions are referenced differently.

ACM_CAP.4.2C

- The evaluator ***shall check*** that the TOE provided for evaluation is labelled with its reference.
    - The evaluator should ensure that the TOE contains a unique reference such that it is possible to distinguish different versions of the TOE. This could be achieved through labelled packaging or media, or by a label displayed by the operational TOE. This is to ensure that it would be possible for consumers to identify the TOE (e.g. at the point of purchase or use).
    - The TOE may provide a method by which it can be easily identified. For example, a software TOE may display its name and version number during the start up routine, or in response to a command line entry. A hardware or firmware TOE may be identified by a part number physically stamped on the TOE.
- The evaluator ***shall check*** that the TOE references used are consistent.
    - If the TOE is labelled more than once then the labels have to be consistent. For example, it should be possible to relate any labelled guidance documentation supplied as part of the TOE to the evaluated operational TOE. This ensures that consumers can be confident that they have purchased the evaluated version of the TOE, that they have installed this version, and that they have the correct version of the guidance to operate the TOE in accordance with its ST. The evaluator can use the configuration list that is part of the provided CM documentation to verify the consistent use of identifiers.

- The evaluator also verifies that the TOE reference is consistent with the ST.
- For guidance on consistency analysis see Annex B.3.

ACM_CAP.4.3C

- The evaluator ***shall check*** that the CM documentation provided includes a configuration list.
  - A configuration list identifies the items being maintained under configuration control.
- The evaluator ***shall check*** that the CM documentation provided includes a CM plan.
- The evaluator ***shall check*** that the CM documentation provided includes an acceptance plan.

ACM_CAP.4.4C

- The evaluator ***shall examine*** the configuration list to determine that it identifies the configuration items that comprise the TOE.
  - The minimum scope of configuration items to be covered in the configuration list is given by ACM_SCP.

ACM_CAP.4.5C

- The evaluator ***shall examine*** the method of identifying configuration items to determine that it describes how configuration items are uniquely identified.

ACM_CAP.4.6C

- The evaluator ***shall check*** that the configuration list uniquely identifies each configuration item.
  - The configuration list contains a list of the configuration items that comprise the TOE, together with sufficient information to uniquely identify which version of each item has been used (typically a version number). Use of this list will enable the evaluator to check that the correct configuration items, and the correct version of each item, have been used during the evaluation.

ACM_CAP.4.7C

- The evaluator ***shall examine*** the CM plan to determine that it describes how the CM system is used to maintain the integrity of the TOE configuration items.
  - The descriptions contained in a CM plan may include:
    - all activities performed in the TOE development environment that are subject to configuration management

procedures (e.g. creation, modification or deletion of a configuration item);

- the roles and responsibilities of individuals required to perform operations on individual configuration items (different roles may be identified for different types of configuration item (e.g. design documentation or source code));
- the procedures that are used to ensure that only authorized individuals can make changes to configuration items;
- the procedures that are used to ensure that concurrency problems do not occur as a result of simultaneous changes to configuration items;
- the evidence that is generated as a result of application of the procedures. For example, for a change to a configuration item, the CM system might record a description of the change, accountability for the change, identification of all configuration items affected, status (e.g. pending or completed), and date and time of the change. This might be recorded in an audit trail of changes made or change control records;
- the approach to version control and unique referencing of TOE versions (e.g. covering the release of patches in operating systems, and the subsequent detection of their application).

ACM_CAP.4.8C

- The evaluator ***shall check*** the CM documentation to ascertain that it includes the CM system records identified by the CM plan.
    - The output produced by the CM system should provide the evidence that the evaluator needs to be confident that the CM plan is being applied, and also that all configuration items are being maintained by the CM system as required by ACM_CAP.4.9C. Example output could include change control forms, or configuration item access approval forms.
- The evaluator ***shall examine*** the evidence to determine that the CM system is being used as it is described in the CM plan.
    - The evaluator should select and examine a sample of evidence covering each type of CM-relevant operation that has been performed on a configuration item (e.g. creation, modification, deletion, reversion to an earlier version) to confirm that all operations of the CM system have been carried out in line with documented procedures. The evaluator confirms that the evidence

includes all the information identified for that operation in the CM plan. Examination of the evidence may require access to a CM tool that is used. The evaluator may choose to sample the evidence.

- For guidance on sampling see Annex B.2.
- Further confidence in the correct operation of the CM system and the effective maintenance of configuration items may be established by means of interview with selected development staff. In conducting such interviews, the evaluator should aim to gain a deeper understanding of how the CM system is used in practice as well as to confirm that the CM procedures are being applied as described in the CM documentation. Note that such interviews should complement rather than replace the examination of documentary evidence, and may not be necessary if the documentary evidence alone satisfies the requirement. However, given the wide scope of the CM plan it is possible that some aspects (e.g. roles and responsibilities) may not be clear from the CM plan and records alone. This is one case where clarification may be necessary through interviews.
- It is expected that the evaluator will visit the development site in support of this activity.
- For guidance on site visits see Annex B.5.

ACM_CAP.4.9C

- The evaluator ***shall check*** that the configuration items identified in the configuration list are being maintained by the CM system.
  - The CM system employed by the developer should maintain the integrity of the TOE. The evaluator should check that for each type of configuration item (e.g. high-level design or source code modules) contained in the configuration list there are examples of the evidence generated by the procedures described in the CM plan. In this case, the approach to sampling will depend upon the level of granularity used in the CM system to control CM items. Where, for example, 10,000 source code modules are identified in the configuration list, a different sampling strategy should be applied compared to the case in which there are only 5, or even 1. The emphasis of this activity should be on ensuring that the CM system is being operated correctly, rather than on the detection of any minor error.
  - For guidance on sampling see Annex B.2.

ACM_CAP.4.10C

- The evaluator ***shall examine*** the CM access control measures described in the CM plan to determine that they are effective in preventing unauthorised access to the configuration items.
    - The evaluator may use a number of methods to determine that the CM access control measures are effective. For example, the evaluator may exercise the access control measures to ensure that the procedures could not be bypassed. The evaluator may use the outputs generated by the CM system procedures and already examined as part of the work unit 4:ACM_CAP.4-13. The evaluator may also witness a demonstration of the CM system to ensure that the access control measures employed are operating effectively.
    - The developer will have provided automated access control measures as part of the CM system and as such their suitability may be verified under the component ACM_AUT.1

ACM_CAP.4.11C

- The evaluator ***shall check*** the CM documentation for procedures for supporting the generation of the TOE.
    - In this work unit the term *generation* applies to those processes adopted by the developer to progress the TOE from implementation to a state acceptable for delivery to the end customer.
    - The evaluator verifies the existence of generation support procedures within the CM documentation. The generation support procedures provided by the developer may be automated, and as such their existence may be verified under the component ACM_AUT.1.2C.
- The evaluator ***shall examine*** the TOE generation procedures to determine that they are effective in helping to ensure that the correct configuration items are used to generate the TOE.
    - The evaluator determines that by following the generation support procedures the version of the TOE expected by the customer (i.e. as described in the TOE ST and consisting of the correct configuration items) would be generated and delivered for installation at the customer site. For example, in a software TOE this may include checking that the procedures ensure that all source files and related libraries are included in the compiled object code.

- The evaluator should bear in mind that the CM system need not necessarily possess the capability to generate the TOE, but should provide support for the process that will help reduce the probability of human error.

ACM_CAP.4.12C

- The evaluator ***shall examine*** the acceptance procedures to determine that they describe the acceptance criteria to be applied to newly created or modified configuration items.
  - An acceptance plan describes the procedures that are to be used to ensure that the constituent parts of the TOE are of adequate quality prior to incorporation into the TOE. The acceptance plan should identify the acceptance procedures to be applied:
    - at each stage of the construction of the TOE (e.g. module, integration, system);
    - to the acceptance of software, firmware and hardware components;
    - to the acceptance of previously evaluated components.
  - The description of the acceptance criteria may include identification of:
    - developer roles or individuals responsible for accepting such

    configuration items;
    - any acceptance criteria to be applied before the configuration items are accepted (e.g. successful document review, or successful testing in the case of software, firmware or hardware).

ACM_SCP.3.1C

- The evaluator ***shall check*** that the configuration list includes the minimum set of items required by the CC to be tracked by the CM system.
  - The list should include the following as a minimum:
    - all documentation required to meet the target level of assurance;
    - test software (if applicable);
    - the TOE implementation representation (i.e. the components or subsystems that compose the TOE). For a software-only TOE, the implementation representation may consist solely of source code; for a TOE that includes a hardware platform, the implementation representation may

refer to a combination of software, firmware and a description of the hardware (or a reference platform).

- the documentation used to record details of reported security flaws associated with the implementation (e.g. problem status reports derived from a developer's problem reporting database).
- **development tools (e.g. programming languages and compilers) and related documentation (e.g. Information pertaining to TOE generation items (such as compiler options, installation/generation options, and build options).**

- **The evaluator shall confirm the level of documentation necessary for ensuring proper visibility of the development tools and related documentation throughout the entire CM documentation.**
- **The evaluator shall confirm the development tools and related documentation listed in the CM documentation is resident in the CM system. In performance of this action item, the evaluator does not need independent access to determine the sufficiency of correspondence between the system configuration list and documentation.**

ACM_SCP.3.2C

- The evaluator ***shall examine*** the CM documentation to determine that the procedures describe how the status of each configuration item can be tracked throughout the lifecycle of the TOE.
  - The procedures may be detailed in the CM plan or throughout the CM documentation. The information included should describe:
    - how each configuration item is uniquely identified, such that it is possible to track versions of the same configuration item;
    - how configuration items are assigned unique identifiers and how they are entered into the CM system;
    - the method to be used to identify superseded versions of a configuration item;
    - the method to be used for identifying and tracking configuration items through each stage of the TOE development and maintenance lifecycle (i.e. requirements specification, design, source code development, through to object code generation and on to executable code, module testing, implementation and operation);

- the method used for assigning the current status of the configuration item at a given point in time and for tracking each configuration item through the various levels of representation at the development phase (i.e. source code development, through to object code generation and on to executable code, module testing and documentation);
- the method used for identifying and tracking flaws relative to configuration items throughout the development lifecycle;
- the method used for identifying correspondence between configuration items such that if one configuration item is changed it can be determined which other configuration items will also need to be changed.

- The analysis of the CM documentation for some of this information may have been satisfied by work units detailed under ACM_CAP.

C. CM GUIDELINES FOR EAL6

The following guidelines incorporate the changes and additions required in order to perform an evaluation at EAL5. Where appropriate, changes made to existing guidance inherited from EAL5 are noted in bold face type.

ACM_AUT.2.1C

- The evaluator ***shall check*** the CM plan for a description of the automated measures to control access to the TOE implementation representation.
- The evaluator ***shall examine*** the automated access control measures to determine that they are effective in preventing unathorised modification of the TOE implementation representation.
- **The evaluator shall check the CM plan for a description of the automated measures to control access to the configuration items.**
- **The evaluator shall examine the automated access control measures to determine that they are effective in preventing unauthorized modification of configuration items.**
- The evaluator reviews the configuration management documentation to identify those individuals or roles authorized to make changes to the TOE implementation representation. For example, once it is under configuration management, access to an element of the implementation representation may only be allowed for the individual who performs the software integration role. **Similarly, the evaluator reviews the CM documentation to identify those individuals or roles authorized to access a configuration item.**

- The evaluator should exercise the automated access control measures to determine whether they can be bypassed by an unathorised role or user. This determination need only comprise a few basic tests. **Tests should cover the TOE implementation representation and configuration items.**

ACM_AUT.2.2C

- The evaluator ***shall check*** the CM documentation for automated means to support generation of the TOE from its implementation representation.
  - In this work unit the term *generation* applies to those processes adopted by the developer to progress the TOE from its implementation to a state ready to be delivered to the end customer.
  - The evaluator should verify the existence of automated generation support procedures within the CM documentation.
- The evaluator ***shall examine*** the automated generation procedures to determine that they can be used to support generation of the TOE.
  - The evaluator determines that by following the generation procedures a TOE would be generated that reflects its implementation representation. The customer can then be confident that the version of the TOE delivered for installation implements the TSP as described in the ST. For example, in a software TOE this may include checking that the automated generation procedures help to ensure that all source files and related libraries that are relied upon to enforce the TSP are included in the compiled object code.
  - It should be noted that this requirement is only to provide support. For example, an approach that placed Unix makefiles under configuration management should be sufficient to meet the aim, given that in such a case automation would have made a significant contribution to accurate generation of the TOE. Automated procedures can assist in identifying the correct configuration items to be used in generating the TOE.

ACM_AUT.2.3C

- The evaluator ***shall check*** that the CM plan includes information on the automated tools used in the CM system.

ACM_AUT.2.4C

- The evaluator ***shall examine*** the information relating to the automated tools provided in the CM plan to determine that it describes how they are used.

- The information provided in the CM plan provides the necessary detail for a user of the CM system to be able to operate the automated tools correctly in order to maintain the integrity of the TOE. For example, the information provided may include a description of:
    - the functionality provided by the tools;
    - how this functionality is used by the developer to control changes to the implementation representation;
    - how this functionality is used by the developer to support generation of the TOE.

- The evaluator ***shall examine*** the CM system to determine that the automated tools and procedures described in the CM plan are used.
    - This work unit may be viewed as an additional activity to be carried out in parallel with the evaluator's examination into the use of the CM system required by ACM_CAP. The evaluator looks for evidence that the tools and procedures are in use. This should include a visit to the development site to witness operation of the tools and procedures, and an examination of evidence produced through their use.
    - For guidance on site visits see Annex B.5.

ACM_AUT.2.5C

- **The evaluator shall check the CM documentation for a description of the automated means by which changes can be ascertained between the TOE and its preceding version.**
- **The evaluator shall examine the CM system to ensure it has the capability to provide an automated means to ascertain the changes between the TOE and its preceding version.**
    - **The evaluator should verify the existence of the automated means to ascertain changes procedures within the CM documentation.**
    - **Previous TOE versions may or may not exist. Regardless, evaluators should exercise the automated mechanisms within the CM system or verify the capability exists.**

ACM_AUT.2.6C

- **The evaluator shall check that the CM documentation includes information on the automated means to identify all other configuration items that are affected by the modification of a given configuration item.**

- **The evaluator shall examine the automated mechanisms used by the CM system to identify all other configuration items that are affected by the modification of a given configuration item.**
    - **The CM documentation should state how the automated means will identify the changes to configuration items that are affected by the modification of a given configuration item. CM systems may differ on how this information is presented.**

ACM_CAP.5.1C

- The evaluator ***shall check*** that the version of the TOE provided for evaluation is uniquely referenced.
    - The evaluator should use the developer's CM system to validate the uniqueness of the reference by checking the configuration list to ensure that the configuration items are uniquely identified. Evidence that the version provided for evaluation is uniquely referenced may be incomplete if only one version is examined during the evaluation, and the evaluator should look for a referencing system that is capable of supporting unique references (e.g. use of numbers, letters or dates). However, the absence of any reference will normally lead to a fail verdict against this requirement unless the evaluator is confident that the TOE can be uniquely identified.
    - The evaluator should seek to examine more than one version of the TOE (e.g. during rework following discovery of a vulnerability), to check that the two versions are referenced differently.

ACM_CAP.5.2C

- The evaluator ***shall check*** that the TOE provided for evaluation is labelled with its reference.
    - The evaluator should ensure that the TOE contains a unique reference such that it is possible to distinguish different versions of the TOE. This could be achieved through labelled packaging or media, or by a label displayed by the operational TOE. This is to ensure that it would be possible for consumers to identify the TOE (e.g. at the point of purchase or use).
    - The TOE may provide a method by which it can be easily identified. For example, a software TOE may display its name and version number during the start up routine, or in response to a command line entry. A hardware or firmware TOE may be identified by a part number physically stamped on the TOE.
- The evaluator ***shall check*** that the TOE references used are consistent.
    - If the TOE is labelled more than once then the labels have to be consistent. For example, it should be possible to relate any labelled

guidance documentation supplied as part of the TOE to the evaluated operational TOE. This ensures that consumers can be confident that they have purchased the evaluated version of the TOE, that they have installed this version, and that they have the correct version of the guidance to operate the TOE in accordance with its ST. The evaluator can use the configuration list that is part of the provided CM documentation to verify the consistent use of identifiers.

- The evaluator also verifies that the TOE reference is consistent with the ST.
- For guidance on consistency analysis see Annex B.3.

ACM_CAP.5.3C

- The evaluator ***shall check*** that the CM documentation provided includes a configuration list.
  - A configuration list identifies the items being maintained under configuration control.
- The evaluator ***shall check*** that the CM documentation provided includes a CM plan.
- The evaluator ***shall check*** that the CM documentation provided includes an acceptance plan.
- **The evaluator *shall check* that the CM documentation provided includes integration procedures.**

ACM_CAP.5.4C

- The evaluator ***shall examine*** the configuration list to determine that it identifies the configuration items that comprise the TOE.
  - The minimum scope of configuration items to be covered in the configuration list is given by ACM_SCP.

ACM_CAP.5.5C

- The evaluator ***shall examine*** the method of identifying configuration items to determine that it describes how configuration items are uniquely identified.

ACM_CAP.5.6C

- The evaluator ***shall check*** that the configuration list uniquely identifies each configuration item.
  - The configuration list contains a list of the configuration items that comprise the TOE, together with sufficient information to uniquely identify which version of each item has been used (typically a version number). Use of this list will enable the evaluator to check

that the correct configuration items, and the correct version of each item, have been used during the evaluation.

ACM_CAP.5.7C

- The evaluator ***shall examine*** the CM plan to determine that it describes how the CM system is used to maintain the integrity of the TOE configuration items.
    - The descriptions contained in a CM plan may include:
        - all activities performed in the TOE development environment that are subject to configuration management procedures (e.g. creation, modification or deletion of a configuration item);
        - the roles and responsibilities of individuals required to perform operations on individual configuration items (different roles may be identified for different types of configuration item (e.g. design documentation or source code));
        - the procedures that are used to ensure that only authorized individuals can make changes to configuration items;
        - the procedures that are used to ensure that concurrency problems do not occur as a result of simultaneous changes to configuration items;
        - the evidence that is generated as a result of application of the procedures. For example, for a change to a configuration item, the CM system might record a description of the change, accountability for the change, identification of all configuration items affected, status (e.g. pending or completed), and date and time of the change. This might be recorded in an audit trail of changes made or change control records;
        - the approach to version control and unique referencing of TOE versions (e.g. covering the release of patches in operating systems, and the subsequent detection of their application).

ACM_CAP.5.8C

- The evaluator ***shall check*** the CM documentation to ascertain that it includes the CM system records identified by the CM plan.
    - The output produced by the CM system should provide the evidence that the evaluator needs to be confident that the CM plan is being applied, and also that all configuration items are being maintained by the CM system as required by ACM_CAP.4.9C.

Example output could include change control forms, or configuration item access approval forms.

- The evaluator ***shall examine*** the evidence to determine that the CM system is being used as it is described in the CM plan.
    - The evaluator should select and examine a sample of evidence covering each type of CM-relevant operation that has been performed on a configuration item (e.g. creation, modification, deletion, reversion to an earlier version) to confirm that all operations of the CM system have been carried out in line with documented procedures. The evaluator confirms that the evidence includes all the information identified for that operation in the CM plan. Examination of the evidence may require access to a CM tool that is used. The evaluator may choose to sample the evidence.
    - For guidance on sampling see Annex B.2.
    - Further confidence in the correct operation of the CM system and the effective maintenance of configuration items may be established by means of interview with selected development staff. In conducting such interviews, the evaluator should aim to gain a deeper understanding of how the CM system is used in practice as well as to confirm that the CM procedures are being applied as described in the CM documentation. Note that such interviews should complement rather than replace the examination of documentary evidence, and may not be necessary if the documentary evidence alone satisfies the requirement. However, given the wide scope of the CM plan it is possible that some aspects (e.g. roles and responsibilities) may not be clear from the CM plan and records alone. This is one case where clarification may be necessary through interviews.
    - It is expected that the evaluator will visit the development site in support of this activity.
        - For guidance on site visits see Annex B.5.

ACM_CAP.5.9C

- The evaluator ***shall check*** that the configuration items identified in the configuration list are being maintained by the CM system.
    - The CM system employed by the developer should maintain the integrity of the TOE. The evaluator should check that for each type of configuration item (e.g. high-level design or source code modules) contained in the configuration list there are examples of the evidence generated by the procedures described in the CM plan. In this case, the approach to sampling will depend upon the level of granularity used in the CM system to control CM items. Where, for example, 10,000 source code modules are identified in

the configuration list, a different sampling strategy should be applied compared to the case in which there are only 5, or even 1. The emphasis of this activity should be on ensuring that the CM system is being operated correctly, rather than on the detection of any minor error.

- For guidance on sampling see Annex B.2.

ACM_CAP.5.10C

- The evaluator ***shall examine*** the CM access control measures described in the CM plan to determine that they are effective in preventing unathorised access to the configuration items.
  - The evaluator may use a number of methods to determine that the CM access control measures are effective. For example, the evaluator may exercise the access control measures to ensure that the procedures could not be bypassed. The evaluator may use the outputs generated by the CM system procedures and already examined as part of the work unit 4:ACM_CAP.4-13. The evaluator may also witness a demonstration of the CM system to ensure that the access control measures employed are operating effectively.
  - The developer will have provided automated access control measures as part of the CM system and as such their suitability may be verified under the component ACM_AUT.1

ACM_CAP.5.11C

- The evaluator ***shall check*** the CM documentation for procedures for supporting the generation of the TOE.
  - In this work unit the term *generation* applies to those processes adopted by the developer to progress the TOE from implementation to a state acceptable for delivery to the end customer.
  - The evaluator verifies the existence of generation support procedures within the CM documentation. The generation support procedures provided by the developer may be automated, and as such their existence may be verified under the component ACM_AUT.1.2C.
- The evaluator ***shall examine*** the TOE generation procedures to determine that they are effective in helping to ensure that the correct configuration items are used to generate the TOE.
  - The evaluator determines that by following the generation support procedures the version of the TOE expected by the customer (i.e. as described in the TOE ST and consisting of the correct configuration items) would be generated and delivered for

installation at the customer site. For example, in a software TOE this may include checking that the procedures ensure that all source files and related libraries are included in the compiled object code.

- The evaluator should bear in mind that the CM system need not necessarily possess the capability to generate the TOE, but should provide support for the process that will help reduce the probability of human error.

ACM_CAP.5.12C

- The evaluator ***shall examine*** the acceptance procedures to determine that they describe the acceptance criteria to be applied to newly created or modified configuration items.
  - An acceptance plan describes the procedures that are to be used to ensure that the constituent parts of the TOE are of adequate quality prior to incorporation into the TOE. The acceptance plan should identify the acceptance procedures to be applied:
    - at each stage of the construction of the TOE (e.g. module, integration, system);
    - to the acceptance of software, firmware and hardware components;
    - to the acceptance of previously evaluated components.
  - The description of the acceptance criteria may include identification of:
    - developer roles or individuals responsible for accepting such configuration items;
    - any acceptance criteria to be applied before the configuration items are accepted (e.g. successful document review, or successful testing in the case of software, firmware or hardware).

ACM_CAP.5.13C

- **The evaluator shall check that the integration procedures describe how the CM system is applied in the TOE manufacturing process.**

ACM_CAP.5.14C

- **The evaluator shall check that the CM documentation describes configuration item acceptance procedures.**
- **The evaluator shall verify configuration item acceptance procedures in the CM system perform as described by the CM documentation.**

- **The CM system should prevent the person who developed the configuration item from being able to accept it.**

ACM_CAP.5.15C

- **The evaluator shall verify that the CM documentation clearly identifies all configuration items that comprise the TSF.**
- **The evaluator shall verify that the configuration items that comprise the TSF are clearly identified in the CM system.**

ACM_CAP.5.16C

- **The evaluator shall verify the CM documentation describes the CM systems audit capabilities.**
  - **The CM system at a minimum will track all TOE modifications to include: the originator, date, and time in the audit trail.**
- **The evaluator shall verify the CM system's TOE audit capabilities.**
  - **The audit portion of the CM system shall provide at a minimum the originator, date and time in the audit trail.**

ACM_CAP.5.17C

- **The evaluator shall verify that the CM documentation describes how the CM system will identify the master copy of all material used to generate the TOE.**
- **The evaluator shall verify that the CM system identifies the master copy of all material used to generate the TOE.**

ACM_CAP.5.18C

- **The evaluator shall verify that the CM documentation describes how the use of the CM system and development security measures will only allow authorized changes to be made to the TOE.**

ACM_CAP.5.19C

- **The evaluator shall verify that the CM documentation describes how the use of the integration procedures will ensure that the generation of the TOE is performed correctly in the authorized manner.**

ACM_CAP.5.20C

- **The evaluator shall verify that the CM documentation sufficiently describes how the CM system will ensure that the person responsible for accepting a configuration item into the CM system is not the person who developed it.**

ACM_CAP.5.21C

- **The evaluator shall verify the CM documentation provides adequate justification that the acceptance procedures provide adequate and appropriate review for all changes made to configuration items.**

ALC_DVS Dependency

- **The evaluator shall confirm that CM-specific security measures from the TSF are implemented and documented as necessary in the CM documentation.**

ACM_SCP.3.1C

- The evaluator ***shall check*** that the configuration list includes the minimum set of items required by the CC to be tracked by the CM system.
    - The list should include the following as a minimum:
        - all documentation required to meet the target level of assurance;
        - test software (if applicable);
        - the TOE implementation representation (i.e. the components or subsystems that compose the TOE). For a software-only TOE, the implementation representation may consist solely of source code; for a TOE that includes a hardware platform, the implementation representation may refer to a combination of software, firmware and a description of the hardware (or a reference platform).
        - the documentation used to record details of reported security flaws associated with the implementation (e.g. problem status reports derived from a developer's problem reporting database).
        - development tools (e.g. programming languages and compilers) and related documentation (e.g. Information pertaining to TOE generation items (such as compiler options, installation/generation options, and build options).
- The evaluator shall confirm the level of documentation necessary for ensuring proper visibility of the development tools and related documentation throughout the entire CM documentation.
- The evaluator shall confirm the development tools and related documentation listed in the CM documentation is resident in the CM system. In performance of this action item, the evaluator does not need independent access to determine the sufficiency of correspondence between the system configuration list and documentation.

ACM_SCP.3.2C

- The evaluator ***shall examine*** the CM documentation to determine that the procedures describe how the status of each configuration item can be tracked throughout the lifecycle of the TOE.
    - The procedures may be detailed in the CM plan or throughout the CM documentation. The information included should describe:
        - how each configuration item is uniquely identified, such that it is possible to track versions of the same configuration item;
        - how configuration items are assigned unique identifiers and how they are entered into the CM system;
        - the method to be used to identify superseded versions of a configuration item;
        - the method to be used for identifying and tracking configuration items through each stage of the TOE development and maintenance lifecycle (i.e. requirements specification, design, source code development, through to object code generation and on to executable code, module testing, implementation and operation);
        - the method used for assigning the current status of the configuration item at a given point in time and for tracking each configuration item through the various levels of representation at the development phase (i.e. source code development, through to object code generation and on to executable code, module testing and documentation);
        - the method used for identifying and tracking flaws relative to configuration items throughout the development lifecycle;
        - the method used for identifying correspondence between configuration items such that if one configuration item is changed it can be determined which other configuration items will also need to be changed.
    - The analysis of the CM documentation for some of this information may have been satisfied by work units detailed under ACM_CAP.

D. CM GUIDELINES FOR EAL7

There are no mandated changes for CM requirements at EAL7. Guidelines for EAL6 are used when performing an evaluation. As mentioned previously, special consideration is needed when determining the sample size.

VI. CONCLUSION AND RECOMMENDATIONS

As a starting point, CEM guidelines for evaluators only suggest minimum activities that should be performed in order to conduct evaluations. As we have seen, higher assurance levels require more evaluator actions in order to achieve the desired confidence that the TOE will perform as intended.

A. CONCLUSION

Webster's Dictionary defines research as [WEB86]:

> Studious inquiry or examination aimed at the discovery and interpretation of facts, revision of accepted theories or laws in the light of new facts, or practical application of such new or revised theories or laws.

As a research document, this work invokes aspects of the above definition when answering the question, "Are Configuration Management (CM) guidelines useful in the evaluation of trusted systems". Studious inquiry of the Common Criteria yielded crucial facts throughout the research process. Part 3 of the Common Criteria [CCP399] provides the complete set of CM (developer and evaluator) requirements for all assurance levels. While a complete set of requirements is available, guidelines for evaluating the accomplishment of the requirements are incomplete.

By stopping at EAL4, the CEM [CEM99] fails to provide CM guidelines needed to ensure a successful product evaluation at high assurance levels (EAL5-7). Lack of guidance leaves requirement interpretation open to individual evaluators. The recommended CM guidelines developed here provide a starting point for the evaluation process and enable uniform application of CM requirements at high assurance levels.

CM guideline recommendations have been made by utilizing the idea of "inheritance and change" provided by the graded assurance scale within the CC. Higher assurance levels build on existing requirements, make changes where needed and add requirements in order to provide increased assurance at the next higher level. EAL4 guidance from the CEM combined with the requirements from the CC provided a key stepping stone for guideline generation. Guidance extension of the CEM for high assurance CM leads to the hypothesis: Configuration Management guidelines for trusted computing systems are useful.

The guidelines recommended in this work pick up at EAL5 where the current CEM provides no guidance. The result is a complete set of guidelines from which several benefits accrue. A complete set of guidelines will be useful in helping developers generate the required content and presentation evidence needed for developing products at any evaluated assurance level. Recommended guidelines will also provide a useful starting point for professional evaluators when performing a CM evaluation at the higher assurance levels.

B. RECOMMENDATIONS

Successful guideline generation does not mean this work is a success. More feedback is needed to verify that the author's interpretation of the Common Criteria [CCP399] is relevant and useful. While it is recognized feedback can come in several forms, appropriate feedback would include comments from professional evaluators. Other feedback could result from using the recommended guidelines to perform an evaluation at EAL5, 6, or 7.

C. FUTURE RESEARCH

Future research which incorporates feedback will enhance the benefits provided from the creation of initial CM evaluation methodology guidelines. Other areas of research which focus on proposed changes to the Common Criteria may also be needed.

1. Change within the Common Criteria

The countries which recognize the Common Criteria met in January this year. The representatives from Germany have proposed a change to the Common Criteria [ONC03]. If approved this change will combine three of the assurance classes that make up the Common Criteria: Configuration Management, Life Cycle Support, and Delivery and Operation. Future study may be needed to ensure that the proposed extension of CEM guidelines for high assurance CM in this work remain relevant regarding future changes within the Common Criteria.

2. Other Regulatory Guidance

Other guidance places information security and assurance requirements on products and systems used and purchased by the Department of Defense. The National Information Assurance Certification and Accreditization Process (NIACAP) [NST00] and the Department of Defense Information Technology Security Certification and

Accreditation Process (DITSCAP) [DCP00] are two examples of other regulatory guidance. DITSCAP and NIACAP contain CM requirements similar to the Common Criteria. Investigation of other regulatory guidance produced after the Common Criteria that provides CM guidance should be conducted to determine where overlap or conflicting requirements may exist. Common CM evaluation guidance derived from the overlap between the Common Criteria, DITSCAP, NIACAP, and other regulatory documents may further enhance the requirements for an IT product evaluation.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

AGT88 — National Computer Security Center. (1988). *A Guide to Understanding Configuration Management in Trusted Systems.* Ft. Meade, Maryland: National Computer Security Center. (NCSC-TG-006-88).

CCP199 — Common Criteria Sponsoring Organizations. (1999). *Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model* (Version 2.1). Retrieved October 2003 from http://csrc.nist.gov/cc/CC-v2.1.html.

CCP399 — Common Criteria Sponsoring Organizations. (1999). *Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance requirements* (Version 2.1). Retrieved October 2003 from http://csrc.nist.gov/cc/CC-v2.1.html.

CEM99 — Common Criteria Sponsoring Organizations. (1999). *Common Methodology for Information Technology Security Evaluation, Part 2: Evaluation Methodology* (Version 1.0). Retrieved October 2003 from http://csrc.nist.gov/cc/CEM.html.

CMP03 — Hass, Anne Mette Jonassen. (2003). *Configuration Management Principles and Practice*. Boston, Massachusetts: Addison-Wesley.

DCP00 — Department of Defense. (2000). *Department of Defense Information Technology Security Certification and Accreditation Process* (DOD 5200.40). Retrieved March 2004 from http://www.iease.disa.mil/ditscap/1.

ECC03 — Roback, Edward. (2003). *Exploring Common Criteria: Can it Ensure that the Federal Government Gets Needed Security in Software?* Retrieved January 2004 from http://www.nist.gov/testimony/2003/erobackcc.html.

ICM96 — Buckley, Fletcher, J. (1996). *Implementing Configuration Management: hardware, software, and firmware* (2nd ed.). Los Alamitos, California: IEEE Computer Society Press.

NST00 — National Security Telecommunication and Information Systems Security Committee. (2000). *National Information Assurance Certification and Accreditization Process* (NSTISSI No. 1000). Retrieved March 2004 from http://www.nstissci.gov/Assets/pdf/sntissi_1000.pdf.

ONC03 — Common Criteria Organization. (2003). *Outline of a new concept for ALC, ACM, and ADO.* Outline provided by Mr. Ron Bottomly.

SCS80 Myers, Philip, A. (1980). *The Neglected Aspect of Computer Security.* Naval Postgraduate School, Monterey, California: Thesis.

SMS03 The Center for Information Systems Security Studies and Research, Naval Postgraduate School. (2003). *Secure Management of Systems* (Course Notes for CS3670).

WEB86 (1986). *Webster's Ninth New Collegiate Dictionary.* Springfield, Massachusettes: Merriam-Webster, Inc.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, VA

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, CA

3. Hugo Badillo
   NSA
   Fort George Meade, MD

4. George Bieber
   OSD
   Washington, DC

5. Ron Bottomly
   NSA
   Fort George Meade, MD

6. RADM Joseph Burns
   Fort George Meade, MD

7. Deborah Cooper
   DC Associates, LLC
   Roslyn, VA

8. CDR Daniel L. Currie
   PMW 161
   San Diego, CA

9. LCDR James Downey
   NAVSEA
   Washington, DC

10. Daniel Faigin
    Aerospace Corporation
    Los Angeles, CA

11. Richard Hale
    DISA
    Falls Church, VA

12. LCDR Scott D. Heller
SPAWAR
San Diego, CA

13. Russell Jones
N641
Arlington, VA

14. Wiley Jones
OSD
Washington, DC

15. David Ladd
Microsoft Corporation
Redmond, WA

16. Dr. Carl Landwehr
National Science Foundation
Arlington, VA

17. Steve LaFountain
NSA
Fort Meade, MD

18. Dr. Greg Larson
IDA
Alexandria, VA

19. Ray A. Letteer
Head, Information Assurance, HQMC C4 Directorate
Washington, DC

20. Penny Lehtola
NSA
Fort Meade, MD

21. Ernest Lucier
Federal Aviation Administration
Washington, DC

22. CAPT Sheila McCoy
Headquarters U.S. Navy
Arlington, VA

23. Dr. Ernest McDuffie
National Science Foundation
Arlington, VA

24. Dr. Vic Maconachy
NSA
Fort Meade, MD

25. Doug Maughan
Department of Homeland Security
Washington, DC

26. John Mildner
SPAWAR
Charleston, SC

27. Dr. John Monastra
Aerospace Corporation
Chantilly, VA

28. Marshall Potter
Federal Aviation Administration
Washington, DC

29. Dr. Roger R. Schell
Aesec
Pacific Grove, CA

30. Keith Schwalm
Good Harbor Consulting, LLC
Washington, DC

31. Dr. Ralph Wachter
ONR
Arlington, VA

32. David Wirth
N641
Arlington, VA

33. Daniel Wolf
NSA
Fort Meade, MD

34. CAPT Robert Zellmann
CNO Staff N614
Arlington, VA

35. Dr. Cynthia E. Irvine
Naval Postgraduate School
Monterey, CA

36. Dr. Nelson J. Irvine
Naval Postgraduate School
Monterey, CA

37. Timothy E. Levin
Naval Postgraduate School
Monterey, CA

38. Dr. Dan C. Boger
Naval Postgraduate School
Monterey, CA

39. LT Michael E. Gross
Naval Postgraduate School
Monterey, CA